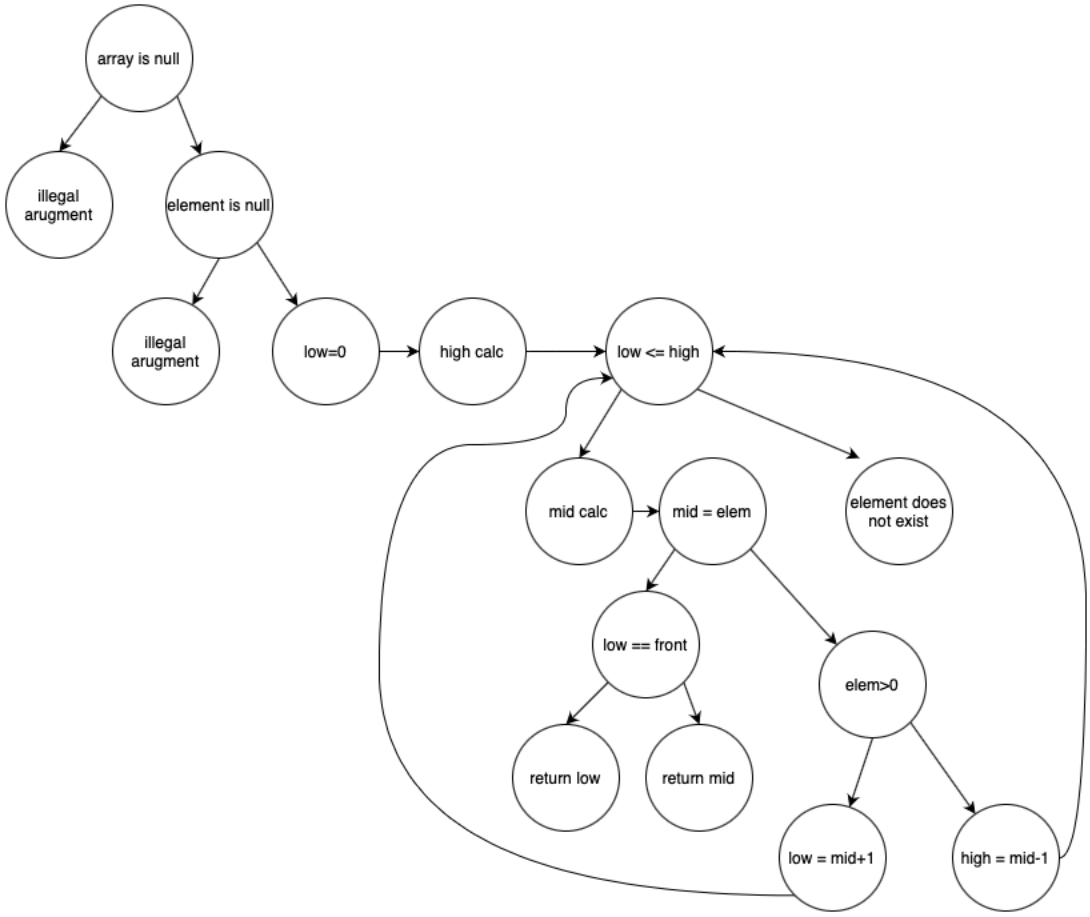
1. For my open-box testing, I chose to only test for Statement Coverage according to the below graph.



I found it would be more time consuming and complicated to write tests to check Branch coverage when I knew I could test for bugs just a thoroughly but easier and with less test through Closed-box testing.

For my Closed-box tests I tested for the following preconditions: the array is not null, the element is not null and the post-condition that that the element was found. I also tested the binary search for a variety of variable types: Integer, Double, and String. For each variable type, I also considered its different properties and how those properties could be exploited to create a bug. This played a factor in deciding what elements I used in my array for each test and in writing specific tests to exploit a property (e.g. Testing that -0.0 double is not seen as equal to 0.0 double in a binary search). My closed-box tests ultimately comprised of Equivalence partitions, boundary value analysis, Internal Boundaries, and data specific edge cases.

Equivalence partitions/ Boundary Value Analysis

- Element at start of array

- Element at end of array
- Element at middle of an even count array
- Element at middle of odd count array

Internal Boundaries

- Integer.MAXVALUE
- Integer.MAXVALUE+1
- Integer.MINVALUE
- Integer.MINVALUE-1
- Double. MAXVALUE
- Double. MAXVALUE+1
- Double. MINVALUE
- Double. MINVALUE-1

Data type specific edge cases

- Tested repeat elements
- Tested null string
- Positive and negative double zeros
- Test if large String size can be found

2. Since the instructions only specified the outcomes for a binary search of a sorted array but checking if an array was sorted would affect the search time, I made the decision for my code to have the ability to take in an unsorted array an attempt a binary search on it. It may or may not return the desired element's location or claim that the element was not found based on an element's location in the array. I tested if the binary search works correctly in my closed-box testing section of my code and it indeed fails as it should. I executed this test in Closed-box Testing because closed box testing is when the tester is only aware of a test's inputs and outputs and not the internal workings and specifications of the code. I could not test the binary search on an unsorted array in open-box testing as I would be aware that one of the specifications is to search only sorted arrays.